European Data Portal

Data Supplier Guidelines

Deliverable Name:	**Data Supplier Requirements**
Status:	**Draft v1.0**
Work Package:	**S1WP3 / S1WP4**
Author(s):	**Intrasoft, Fraunhofer Fokus, con terra, Sogeti**
Partner(s) contributing:	
Date:	**5 November 2015**

EDP_S1_GDL_Data-Supplier-Guidelines_v1

Table of Contents

1 Reference Documents

This section lists the documents which are referenced within those Supplier Guidelines.

Id	Title
[1]	Technical Guidance for the implementation of INSPIRE Discovery Services, Initial Operating Capability Task Force for Network Services, 07-11-2011
[2]	INSPIRE Metadata Implementing Rules: Technical Guidelines based on EN ISO 19115 and EN ISO 19119, European Commission Joint Research Centre, 16-06-2010
[3]	OGC Catalogue Services Specification 2.0.2 - ISO Metadata Application Profile, Version 1.0, OGC doc 07-045
[4]	OpenSearch, http://www.opensearch.org/Specifications/OpenSearch/1.1
[5]	OGC OpenSearch GeoSpatial and Temporal Extensions, version: 1.0.0, OGC 10-032r6

Table 1-1: Reference Documents

2 Introduction

The **European Data Portal** (EDP) began by harvesting national (open) data portals. Progressively, it aims at harvesting more and more data portals share public sector information in an open manner.

If you wish that your portal or website is harvested by the European Data Portal, the questions below will guide you through to the next steps:

1. *Are you publishing public sector information?*

☐ No.

☐ Yes, please go to the next question.

2. *Are you already being harvested by another national or local portal?*

☐ Yes, which one? _____

☐ No, please go to the next question.

3. *Is your national Open Data Portal harvesting INSPIRE compliant catalogues?*

☐ Yes – the European Data Portal will also harvest them

☐ No

If no, do you want the European Data Portal to set up a mapping process that takes INSPIRE compliant metadata (ISO 19139) that is present in your national geo-catalogue?

☐ No

☐ Yes, please go the next section where the technical requirements will be explained.

To be harvested by the European Data Portal we have several technical/operational requirements we wish to share with you.

These requirements are described/referenced here:

- Technical requirements/constraints (link)
- Operational requirements (link)

If you want to learn more about how to improve your open data publishing, please consult the different sections of the EDP website:

- Providing data - Goldbook (link)
- Providing training (link)

3 Purpose of this document

The objective of this document is to identify and describe the requirements that data suppliers (e.g. national portals, public data portals in the EU Member States, portals from international organizations etc.) must fulfill for being harvested by the European Data Portal.

This document is not a Service Level Agreement (SLA) between a data supplier and the European Data Portal rather than a description of the general and technical requirements/constraints, the operational process and the governance required for an effective and efficient harvesting of the datasets (metadata) from the data supplier.

The checklist in section 6 may serve as a quick reference for these requirements and settings that have to be implemented on the data supplier site.

4 Technical Requirements/Constraints

4.1 Overview of the Harvesting Process

(overview of the harvesting process to be included here)

4.2 Access to harvested sites

4.2.1 Authentication

Some source sites require authentication of the harvesting site in the form of an account with login name and password before being able to be harvested by another site (here the EDP).

Hereto the EDP responsible will contact the source site responsible and request an account / sign-up for harvesting the source site by the EDP.

4.2.2 API access to harvested site

In order for the EDP harvesting process to harvest datasets from a source site, the latter needs to implement one of the interfaces as described in detail in section 3.3 below.

4.2.3 FTP Access to harvested site

The European Data Portal currently does not support FTP for downloading datasets from a source site.

4.2.4 Pushing Datasets to the EDP Metadata Repository (**not yet available**)

In case that a data supplier does not provide an API or any other interface (endpoint) from which the datasets could be harvested by the EDP harvester modules, the data supplier can also push the datasets into the EDP's CKAN-based metadata repository. This process needs to be agreed between the data supplier and the Portal project responsible prior to any enabling of the CKAN interface.

4.3 Interfaces supported for harvesting

The following sections describe **the list of interfaces** that data suppliers (e.g. national portals, public data portals in the Member States, portals from international organizations etc.) must have in place for being harvested by the European Data Portal.

The main supported interfaces are the following:

- **DCAT-AP / CKAN** compliant sites (for "normal" datasets)
- **INSPIRE** Catalogue Services (for geospatial datasets)
- **OpenSearch** (GEO/EOP) (for geospatial datasets)

4.3.1 DCAT-AP

Providing data via a DCAT-AP interface is the official recommend method.

4.3.1.1 General remarks

DCAT-AP is a specification for describing public sector datasets in Europe. It's based on the Data Catalogue vocabulary[1]. The datasets are provided as linked data and can be represented in multiple ways. For the harvesting process, it is mandatory that an RDF/XML representation is provided.

4.3.1.2 Metadata Model

For general information on the metadata model, please refer to the official documentation[2]. The respective qualifiers (mandatory, recommended and optional) need to be adhered to. Following is an example dataset with all the mandatory properties.

```
<dcat:Dataset>
        <dct:title xml:lang="en">v394d2705_emilia</dct:title>
        <dct:description xml:lang="en">Place: Emilia area; Orbit type:
Descending;  Map type: Mean Ground Velocity; Number of Images: 1; Track:
394; Frame: 2705; SLC: 37; Master: 2008-05-19; Number of interferograms:
139; Processor Name: SBAS algorithm (SARScape); Parameter Name: n.a.;
Satellite: ASAR ENVISAT; Input File Name: n.a.;</dct:description>
        <dct:spatial>
                <dct:Location>
                        <locn:geometry
rdf:datatype="http://www.opengis.net/ont/geosparql#wktLiteral">POLYGON((10.
87649 45.23888,12.38418 45.23888,12.38418 44.144384,10.87649
44.144384,10.87649 45.23888))</locn:geometry>
                </dct:Location>
        </dct:spatial>
        <dct:temporal>
                <dct:PeriodOfTime>
                        <schema:startDate
rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2003-02-
10</schema:startDate>
```

[1] http://www.w3.org/TR/vocab-dcat/
[2] https://joinup.ec.europa.eu/asset/dcat_application_profile

```
                        <schema:endDate
rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2010-06-
09</schema:endDate>
                </dct:PeriodOfTime>
        </dct:temporal>
        <dct:created
rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2012-12-
28</dct:created>
        <dct:conformsTo rdf:parseType="Resource">
                <dct:title xml:lang="en">Corrigendum to INSPIRE Metadata
Regulation published in the Official Journal of the European Union, L 328,
page 83</dct:title>
                <dct:modified
rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2009-12-
15</dct:modified>
        </dct:conformsTo>
        <dcat:theme rdf:resource="http://inspire.ec.europa.eu/theme/ge"/>
        <dcat:theme rdf:resource="http://eurovoc.europa.eu/100151"/>
        <dcat:theme
rdf:resource="http://publicdata.eu/def/category/science-and-technology"/>
        <dcat:theme rdf:resource="http://eurovoc.europa.eu/1151"/>
        <dcat:keyword xml:lang="en">GEOMORPHOLOGY</dcat:keyword>
        <dct:language
rdf:resource="http://publications.europa.eu/resource/authority/language/ENG
"/>
        <dct:provenance>
                <dct:ProvenanceStatement>
                        <rdfs:label xml:lang="en">Validation Flag:
validated; Validation Time: 2013-01-12; Validation Responsible: Dr.
Cristiano Tolomei (cristiano.tolomei@ingv.it)</rdfs:label>
                </dct:ProvenanceStatement>
        </dct:provenance>
        <dcat:landingPage>
                <foaf:Document
rdf:about="http://185.12.7.250:8080/petascope/...">
                        <dct:title
xml:lang="en">ASAR_EMILIADISPDESMEAN_32632_90</dct:title>
                        <dct:description xml:lang="en">WCS request to
download the coverage (raster, geotiff).</dct:description>
                </foaf:Document>
        </dcat:landingPage>
        <dcat:landingPage>
                <foaf:Document
rdf:about="http://185.12.7.250:8080/petascope/...">
                        <dct:title
xml:lang="en">ASAR_EMILIADISPDESMEAN_32632_90</dct:title>
                        <dct:description xml:lang="en">WCS request to
download the coverage (csv).</dct:description>
                </foaf:Document>
        </dcat:landingPage>
        <dct:identifier>VELISAR2.0_emilia_v394d2705</dct:identifier>
</dcat:Dataset>
```

4.3.1.3 Categorization

The EDP uses a controlled vocabulary for categories. Below, you find the categories used in the EDP:

CATEGORY	INTERNAL NAME OF CATEGORY
AGRICULTURE, FISHERIES, FORESTRY, FOODS	agriculture-fisheries-forestry-food
EDUCATION, CULTURE AND SPORT	education-culture-and-sport
ENVIRONMENT	environment
ENERGY	energy
TRANSPORT	transport
SCIENCE AND TECHNOLOGY	science-and-technology
ECONOMY AND FINANCE	economy-and-finance
POPULATION AND SOCIAL CONDITIONS	population-and-social-conditions
HEALTH	health
GOVERNMENT, PUBLIC SECTOR	government-public-sector
REGIONS; CITIES	regions-cities
JUSTICE, LEGAL SYSTEM, PUBLIC SAFETY	justice-legal-system-public-safety
INTERNATIONAL ISSUES	international-issues

When providing data, either these categories should be used or a mapping of the used to categories to these categories should be provided.

4.3.1.4 Requests

The harvester currently supports harvesting from an Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH)[3] compliant source or from reading a dump file containing the RDF/XML representation of the datasets.

For OAI-PMH-compliant sources, only the verb "ListRecords" is used.

4.3.1.5 Responses

As indicated above, the response must be DCAT-AP-compliant to be understood by the harvesting component.

4.3.1.6 Error Handling

The OAI-PMH protocol provides methods for error handling that the harvester can understand. If using this protocol, these error methods should be used.

4.3.1.7 Service Information for Integration

As stated above, a categorization mapping should be provided. Apart from that, the URL for the OAI-PMH endpoint or the dump file is needed.

4.3.2 CKAN API

The open-source data portal platform CKAN[4] is widely used for building Open Data platforms. Its RPC-style[5] API (Action API) is supported as an interface for data suppliers of the European Data Portal. Basically the following options for using that interface are available:

[3] https://www.openarchives.org/pmh/

- The data supplier uses CKAN for providing its Open Data metadata. It is important that the used CKAN version supports the Action API[6]. The legacy APIs of CKAN are not supported.
- The data supplier offers a CKAN compliant API, where the necessary endpoints reproduce the exact API behaviour.

4.3.2.1 Requests and Responses

Only the "package_search" API endpoint is needed in order to harvest the metadata. Its specifications are described in detail in the official documentation[7]. This endpoint is used to get the metadata in a paginated way. Therefore it accepts query parameters in a request and returns a dictionary with datasets as a result. The highlevel use of this endpoint has to be offered as specified in the CKAN documentation.

Example Call: `GET http://www.example.com/api/3/action/package_search?rows=50`

4.3.2.2 Metadata Model

Although the CKAN API can be used as is, the basic CKAN data schema was extended and modified to meet the requirements of the underlying data structure (DCAT-AP) of the European Data Portal. The response of the "package_search" action exposes a "results" field, which holds a list of dictized datasets. The data structure of such a dataset differs from the one of a plain CKAN installation.

Notes:

- Bold fields are CKAN standard. Further information in the official documentation.
- Type specifications according to official JSON standard: http://json.org/.
- **Besided the mandatory fields, the field names and types are not strict, but a data supplier has to make sure an obvious mapping is possible.**
- For a detailed explanation of each field, refer to the DCAT-AP specifications.

Dataset Schema

The following fields are mandatory:

Field	Type	DCAT-AP Dataset equivalent
title	string	dct:title
notes	string	dct:description

[4] http://ckan.org/
[5] Remote Procedure Call
[6] http://docs.ckan.org/en/ckan-2.4.0/api/index.html#action-api-reference
[7] http://docs.ckan.org/en/ckan-2.4.0/api/index.html#ckan.logic.action.get.package_search

The following fields are optional but **highly recommened**:

Field	Type	DCAT-AP Dataset equivalent
contact_point	array of objects (allowed members: type, name, email, resource)	dcat:contactPoint
tags	array of objects	dcat:keyword
publisher	object	dct:publisher
groups	array of objects - **the name of each group needs to fit the official categorization (see 4.3.1.2)**	dcat:theme
resources	array of objects (See **Distribution Schema**)	dcat:distribution

The following fields are optional:

Field	Type	DCAT-AP Dataset equivalent
conforms_to	array of objects (allowed members: label, resource)	dct:conformsTo
creator	object	dct:creator
accrucal_periodicity	object	dct:accrucalPeriodicity
identifier	object	dct:identifier
url	string	dcat:landing_page
language	array of objects (allowed members: label, resource)	dct:language
other_identifier	object	adms:identifier
issued	string	dct:issued
dcat_spatial	array of objects (allowed members: label, resource)	dct:spatial
temporal	array of objects (allowed members: start_date, end_date)	dct:temporal
modified	string	dct:modified
version_info	string	owl:versionInfo

version_notes	string	adms:versionNotes
provenance	array of objects (allowed members: label, resource)	dct:provenance
source	array of strings	dct:source
access_rights	object	dct:accessRights
has_version	array of strings	dct:hasVersion
is_version_of	array of strings	dct:isVersionOf
relation	array of strings	dct:relation
page	array of strings	foaf:page
sample	array of strings	adms:sample
dct_type	string	dct:type

Distribution Schema

The following fields are mandatory:

Field	Type	DCAT-AP Distribution equivalent
url	string	dcat:accessURL

The following fields are optional but **highly recommened**:

Field	Type	DCAT-AP Distribution equivalent
description	string	dct:description
format	string	dct:format
license	object	dct:license

The following fields are optional:

Field	Type	DCAT-AP Distribution equivalent
checksum	object	spdx:checksum
mimetype	string	dcat:mediaType
download_url	array of strings	dcat:downloadURL

issued	string	dct:issued
status	object	adms:status
name	string	dct:title
modified	string	dct:modified
rights	object	dct:rights
page	array of strings	foaf:page
size	number	dcat:byteSize
language	array of objects	dct:language
conforms_to	array of objects	dct:conformsTo

4.3.2.3 Example

A result of the "package_search" action looks like this:


```
{
  "help": "http://example.eu/data/api/3/action/help_show?name=package_search",
  "success": true,
  "result": {
    "count": 113948,
    "sort": "score desc, metadata_modified desc",
    "facets": {},
    "results": [
      {
        "issued": "2011-10-20T00:00:00Z",
        "id": "525abe30-ef60-4bf9-824e-916368c1fad8",
        "metadata_created": "2015-09-15T12:08:54.860742",
        "metadata_modified": "2015-09-15T13:17:51.405474",
        "temporal": [
          {
            "start_date": "2011-10-19T22:00:00Z",
            "end_date": "2011-10-19T22:00:00Z"
          }
        ],
        "state": "active",
        "type": "dataset",
        "resources": [
          {
            "package_id": "525abe30-ef60-4bf9-824e-916368c1fad8",
            "id": "7166a1fa-d994-4d88-8e76-3378930b1e16",
            "state": "active",
            "format": "XHTML",
            "mimetype": "application/xhtml+xml",
            "name": "Example",
            "created": "2015-09-15T14:39:43.865240",
            "url": "http://example.com"
          }
        ],
        "tags": [
          {
            "vocabulary_id": null,
            "state": "active",
            "display_name": "Example Tag",
            "id": "06993102-a2ee-4e40-b9e4-ed3e4b86e943",
            "name": "example-tag"
          }
        ],
        "groups": [
```

```
          {
            "display_name": "Economy and finance",
            "description": "",
            "title": "Economy and finance",
            "id": "128d0956-4526-440e-a951-f153c190d890",
            "name": "economy-and-finance"
          }
        ],
        "creator_user_id": "0ab3c2ec-c2a2-4eef-b70f-ed093e028063",
        "publisher": {
          "resource": "http://example.com "
        },
        "organization": {
          "description": "Example Organization",
          "created": "2015-09-15T13:56:32.985936",
          "title": "Example Organization",
          "name": "example-orag",
          "is_organization": true,
          "state": "active",
          "image_url": "",
          "revision_id": "ea70fb1f-29a8-4e7b-8527-809e4792a75b",
          "type": "organization",
          "id": "0897b420-3c3d-4a19-9c2c-a9815e2db2be",
          "approval_status": "approved"
        },
        "name": "example-dataset",
        "notes": "Example",
        "owner_org": "0897b420-3c3d-4a19-9c2c-a9815e2db2be",
        "modified": "2011-10-20T00:00:00Z",
        "url": "",
        "title": "Example Dataset",
        "identifier": [
          "http://example-ident.com"
        ]
      }
    ],
    "search_facets": {}
  }
}
```


4.3.2.4 Translation

The following fields of datasets and distributions are available in 24 languages:

- title
- description

A data supplier has to make sure, that those field are provided in **English**. It is possible to provide the fields in another language by indicating it with a respective member within the dataset object:


```
"translation_meta": {
   "default": "fr"
}
```


Use Use ISO 639-1 language codes for defining the default language. In addition it is possible to provide already existing translations by adding the following member, either to the dataset object and/or the distribution objects:


```
"translation": {
  "fr": {
    "title": "Title in French",
    "description": "Description in French"
  },
  "es": {
    "title": "Title in Spanish",
    "description": "Description in Spanish"
  }
}
```


Provide vor each language a member, where the name is a valid ISO 639-1 code.

4.3.3 INSPIRE Catalogue Services (for geospatial metadata)

4.3.3.1 General remarks

This interface represents an INSPIRE compliant Catalogue (Discovery) Service [1]. It is defined as a slightly extended version of the OGC CSW AP ISO [3].

The GetCapabilities operation (mandatory for all OGC Services) is not needed for running the harvesting. But this operation could be helpful upon registration of the Catalogue Service within the EU Data Portal as the service´ response provides additional information which must otherwise be found out during the registration (e.g. the supported protocol bindings or the support of the "modified" queryable for selective harvesting).

For the harvesting process just the GetRecords operation will be called. The GetRecordById is not needed.

Operation	Operation Description	EDP usage
GetCapabilities	Retrieve catalog service metadata	Only for gathering service information upon registration
GetRecords	Retrieval of a bunch of metadata items.	Yes
GetRecordById	Retrieval information of single metadata items.	No

Table 2 OGC CSW Operations used by EDP

4.3.3.2 Metadata Model

The metadata model considered is as defined in [1] and [2].

Within a GetRecords query (constraint) just the following metadata model elements (queryables) are used (see Table 3).

Request Parameter	Definition [a]	Used Values in EDP	XPath [b]
Type	Provides the desired information resources.	Always the followoing fixed values used: "dataset", "datasetcollection" and "series"	/gmd:MD_Metadata/gmd:hierarchyLevel/gmd:MD_ScopeCode/@codeListValue
Modified	The metadata datestamp in case of selective harvesting (if supported), see below.	Date	/gmd:MD_Metadata/gmd:dateStamp/gco:Date
a: "Definition" represents the semantic meaning of element in EDP. It is slightly different from the genetic			

meaning in OGC CSW. b: Element's XML Path in GetRecords Request.

Table 3 Table of GetRecords Queryables (not Parameters – see below)

Example query (constraint):

```
<Constraint version="1.1.0">
        <ogc:Filter>
                <ogc:Or>
                        <ogc:PropertyIsEqualTo>
                                <ogc:PropertyName>Type</ogc:PropertyName>
                                <ogc:Literal>dataset</ogc:Literal>
                        </ogc:PropertyIsEqualTo>
                        <ogc:PropertyIsEqualTo>
                                <ogc:PropertyName>Type</ogc:PropertyName>
                                <ogc:Literal>datasetcollection</ogc:Literal>
                        </ogc:PropertyIsEqualTo>
                        <ogc:PropertyIsEqualTo>
                                <ogc:PropertyName>Type</ogc:PropertyName>
                                <ogc:Literal>series</ogc:Literal>
                        </ogc:PropertyIsEqualTo>
                </ogc:Or>
        </ogc:Filter>
</Constraint>
```

As defined in [1] the operation must be able to return ISO19139 metadata aligned with the INSPIRE regulations [2].

4.3.3.3 Requests

The mandatory GetRecords operation works as the primary means of metadata item discovery with HTTP protocol binding. It executes an inventory search and returns the metadata items. Only OGC Filter XML encoding is supported. For the GetRecords requests a few additional requirements exists. These will be explained in the following.

Bindings

As bindings one or more of HTTP POST/XML, POST/XML/SOAP1.1 and POST/XML/SOAP1.2 have to be supported.

Operation Parameters

The following parameters (not the queryables) and parameter values are used in EDP for the GetRecords requests.

Request Parameter	Definition [a]	Used Values in EDP	XPath [b]
service	Tells this is a CSW service.	Always fixed value: CSW	/GetRecords@service
version	Tell which version of CSW service is requested.	Always fixed value; 2.0.2	/GetRecords@version

resultType	Specifies the type of result	Always fixed value: "results"	/GetRecords@resultType
outputFormat	Specifies the output format of GetRecords returned document	Always fixed value: "application/xml"	/GetRecords@outputFormat
outputSchema	Specifies the schema of GetRecords returned document	Always fixed value (namespace): "http://www.isotc211.org/2005/gmd"	/GetRecords@outputSchema
startPosition	Specifies the sequence number of first returned record	Used: integer between 1 and returned number Default value is 1	/GetRecords@startPosition
maxRecords	Specifies number of returned records	Supported: positive integer between 1 and N. Default value is: 50	/GetRecords@maxRecords
typeNames	Specifies the query- and elementSetName type	Always fixed value: "gmd:MD_Metadata" "gmd" is valid namespace prefix for "http://www.isotc211.org/2005/gmd"	/GetRecords/Query@typeName And /GetRecords/Query/ElementSetName@typeName
ElementSetName	Specifies the type of GetRecords returned document	As only full metadata sets will be requested by the Harvester this parameter will always be set to "full".	/GetRecords/Query/ElementSetName
a: "Definition" represents the semantic meaning of element in EDP. It is slightly different from the genetic meaning in OGC CSW. b: Element's XML Path in GetRecords Request.			

Table 4 Table of GetRecords Request Parameters

Partitioning

For partitioning (pagination) the following parameters are used (see Table 4):

- **startPosition**

- **maxRecords**

Selective Harvesting

Selective harvesting allows harvesters to limit harvest requests to just those portions of the metadata available from a repository which have been changed within a specified time frame.

Selective harvesting often makes sense as this would require to harvest only a few metadata records daily as only a few metadata records are gets changed within a day.

For selective harvesting the predefined queryable (usually "modified" – see Table 3) is used.

4.3.3.4 Responses

As defined by [1] the operation must be able to return ISO19139 metadata aligned with the INSPIRE regulations [2].

Partitioning

For partitioning (pagination) as part of the search response, it is **mandatory** to have the total count of matching metadata items returned, even if the metadata for these items is not contained in the search response. This parameter, coupled with the ability to specify the startPosition and the number of desired records (maxRecs) from the Catalogue (see 0), will allow EDP to implement results paging and reducing the load on both the EDP system and on the data partners.

4.3.3.5 Error Handling

Useful status and error messages help the EDP manage client sessions effectively. Any limitations on submitted search requests to the inventory systems should be noted in the response (e.g., "too many records requested", "search timed out") so that predictable error-handling can be managed by the EDP.

4.3.3.6 Service Information for Integration

To be able to integrate an INSPIRE_DS [1] the following information need to be provided by the data supplier:

Service Information	Definition [a]	Obligation (M=Mandatory, O=Optional, C=Conditional)	Datatype
GetRecords URL	URL of the CSW GetRecords operation	M	URL
GetRecords Binding	URL of the CSW GetRecords operation	M	Codelist (one of): "POST/XML", "POST/XML/SOAP1.1" "POST/XML/SOAP1.2"

Modified[a]	Name of the queryable (if supported) for filtering on the metadata datestamp (for selective harvesting)	**Possibly for future use**	String. [Namespace":"]QueryableName
MaxRecordsMax	Specifies the maximal number of maximal returned records	**Possibly for future use (currently always set to "50")**	Integer
[a] = Value in CSW filter will be formatted as "MM-DD-YYYY". Operators: ">=", "<=" will be used.			

Table 5 Service Information needed for Integration

4.3.4 OpenSearch(GEO/EOP) (for geospatial metadata)

4.3.4.1 General remarks

This interface is based on the OpenSearch Service as defined in [4] and in [5] (with some restrictions and a few additional requirements as defined below).

Usually search clients use the XML based OpenSearch description documents (OSDD) to learn about the public interface of a search engine. So they represent the service metadata. The OSDD contains information about the search engine including the parameterized URL templates that indicate how the search client should make search requests (the request interface). Those illustrate the parameters accepted by the service for a variety of output formats (in which results can be obtained).

For EDP harvesting the OSDD is not needed. But it could be helpful upon registration of the Catalogue Service within the EU Data Portal as it provides additional information which must otherwise be found out during the registration (e.g. the required URL template). What is needed is just the URL template for the application/atom+xml content format. A URL template is simple, consisting of a description of a HTTP GET request with a series of usually optional key-value parameters that can be used to constrain the search.

Operation	Operation Description	EDP usage
URL to Open Search Desription Document (OSDD)	Alows to retrieve the Open Search Desription Document prodividing the service metadata	Optional, only for gathering service information upon registration
Request (URL template)	Retrieval of metadata items.	Yes

Table 6 OpenSearch Operations considered by EDP

4.3.4.2 Metadata Model

The metadata model considered is as defined in [4] and [5]. An important point is that ISO19139(-2) metadata must be included in every ATOM entry as described in [5].

Within a request just the following metadata model elements (queryables) are used (see Table 7).

Request Parameter	Definition [a]	Used Values in EDP	XPath [b]
type	Provides the desired information resources. Possible values:	**Possibly for future use (currently not considered)**	/gmd:MD_Metadata/gmd:hierarchyLevel/gmd:MD_ScopeCode/@codeListValue

	"dataset", "datasetcol-lection" and "series"		

a: "Definition" represents the semantic meaning of element in EDP.

b: Element's XML Path in ISO19115.

Table 7 Table of Request Queryables (not Parameters – see below)

4.3.4.3 Requests

The Requests (created from the URL template) which discover the metadata items are based on the HTTP/GET protocol binding. They execeute an inventory search and return the metadata items. Only name(key)-value-pairs (KVP) are supported.

Operation Parameters

The following parameters (not the queryables) and parameter values are used in EDP for the requests. Any other parameter which is not optional must have assigned a fixed value.

Request Parameter	Definition [a]	Used Values in EDP
startIndex	Specifies the sequence number of first returned record. It is **conditional to startPage**	Used: integer between 1 and returned number Default value is 1
startPage	page number of the set of search results desired by the search It is **conditional to startIndex**	Used: integer Default value is 1
count	Specifies number of returned records.	**This parameter will not be set by the harvester (it must be fixed in the URL template or must have a default value).** Usually the value should be 50
recordSchema	The concrete record schema to be returned.	**This parameter will not be set by the harvester (it must be fixed in the URL template (usually "iso") or have a default value which provides ISO19139 format included in an Atom entry)**

Table 8 Request Parameters

Partitioning

For partitioning (pagination) the following parameters are used (see Table 8):

- **startIndex** or **startPage**[8]

Example Request

http://fedeo.esa.int/opensearch/request/?httpAccept=application/atom+xml&type=collection&startIndex=1&count=10&recordSchema=iso

4.3.4.4 Responses

As defined by [OpenSearchEOP] the operation must be able to return ISO19139 metadata embedded as additional metadata with an atom entry.

Such an atom entry may look like this (consider element "gmd:MD_Metadata"):

```
<entry xml:lang="en-US">
<id>http://fedeo.esa.int/opensearch/request/?httpAccept=application%2Fatom%2Bxml&amp;type=collection&amp;startRecord=1&amp;maximumRecords=10&amp;recordSchema=iso&amp;uid=1860_1993_2050_NITROGEN</id>
      <title>GLOBAL MAPS OF ATMOSPHERIC NITROGEN DEPOSITION, 1860, 1993, AND 2050</title>
    <dc:identifier>1860_1993_2050_NITROGEN</dc:identifier>
    <updated>2015-08-12T16:14:08Z</updated>
    <dc:date>1860-01-01/2050-12-31</dc:date>
    <georss:polygon>-90.0 -180.0 90.0 -180.0 90.0 180.0 -90.0 180.0 -90.0 -180.0</georss:polygon>
    <published>2006-04-14T00:00:00Z</published>
    <category label="EARTH SCIENCE > ATMOSPHERE > ATMOSPHERIC CHEMISTRY > NITROGEN COMPOUNDS > NITROGEN OXIDES" term="EARTH SCIENCE > ATMOSPHERE > ATMOSPHERIC CHEMISTRY > NITROGEN COMPOUNDS > NITROGEN OXIDES"/>
    <category label="NH3" term="NH3"/>
    <category label="TM3" term="TM3"/>
    <category label="NH4+" term="NH4+"/>
          …
    <summary type="html"><![CDATA[<table xmlns="">
<tr valign="top">
<td>
<b>Title </b>
</td>
<td>GLOBAL MAPS OF ATMOSPHERIC NITROGEN DEPOSITION, 1860, 1993, AND 2050</td>
</tr>
<tr valign="top">
<td>
<b>Description </b>
</td>
<td>This data set provides global gridded estimates of atmospheric deposition of total inorganic nitrogen (N), NHx (NH3 and NH4+), and NOy (all oxidized forms of nitrogen other than N2O), in mg N/m2/year, for the years 1860 and 1993 and projections
……
</tr>
</table>]]></summary>
    <link href="http://fedeo.esa.int/opensearch/request?httpAccept=application/atom%2Bxml&amp;startRecord=1&amp;parentIdentifier=EOP:ESA:FEDEO&amp;uid=1860_1993_2050_NITROGEN&amp;recordSchema=iso" rel="alternate" title="Atom format" type="application/atom+xml"/>
    <link href="http://fedeo.esa.int/opensearch/description.xml?parentIdentifier=1860_1993_2050_NITROGEN" rel="search" type="application/opensearchdescription+xml"/>

>>>>>>
    <gmd:MD_Metadata xmlns:gmd="http://www.isotc211.org/2005/gmd" xmlns:dif="http://gcmd.gsfc.nasa.gov/Aboutus/xml/dif/" xmlns:fn="http://www.w3.org/2005/02/xpath-functions" xmlns:gco="http://www.isotc211.org/2005/gco" xmlns:geonet="http://www.fao.org/geonetwork" xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:util="java:java.util.UUID">
      <gmd:fileIdentifier>
        <gco:CharacterString>1860_1993_2050_NITROGEN</gco:CharacterString>
      </gmd:fileIdentifier>
      <gmd:language>
        <gco:CharacterString>eng</gco:CharacterString>
```

[8] **count** is not required: it is up to the service how it implements this. E.g. can be missing if startPage is used. It is opaque to the harvester since it is already part of the URL.

```
      </gmd:language>
      <gmd:characterSet>
        <gmd:MD_CharacterSetCode
codeList="http://www.isotc211.org/2005/resources/Codelist/gmxCodelists.xml#MD_CharacterSetCode"
codeListValue="utf8">utf8</gmd:MD_CharacterSetCode>
      </gmd:characterSet>
      <gmd:hierarchyLevel>
        <gmd:MD_ScopeCode codeList="http://www.isotc211.org/2005/resources/Codelist/gmxCodelists.xml#MD_ScopeCode"
codeListValue="series">series</gmd:MD_ScopeCode>
      </gmd:hierarchyLevel>
      <gmd:contact>
        <gmd:CI_ResponsibleParty>
          <gmd:individualName>
            <gco:CharacterString>ORNL DAAC USER SERVICES OFFICE,</gco:CharacterString>
          </gmd:individualName>
          <gmd:contactInfo>
            <gmd:CI_Contact>
              <gmd:phone>
                <gmd:CI_Telephone>
                  <gmd:voice>
                    <gco:CharacterString>(865) 574-7447</gco:CharacterString>
                  </gmd:voice>
                </gmd:CI_Telephone>
              </gmd:phone>
              <gmd:address>
            .....
    </gmd:MD_Metadata>
  </entry>
```

Partitioning

For partitioning (pagination) as part of the search response, it is **mandatory** to have the totalResults of matching metadata items returned, even if the metadata for these items is not contained in the search response. This parameter, coupled with the ability to specify the startIndex or startPage and the number of desired records (count) from the Catalogue, will allow EDP to implement results paging and reducing the load on both the EDP system and on the data partners.

4.3.4.5 Error Handling

Useful status and error messages help the EDP manage client sessions effectively. Any limitations on submitted search requests to the inventory systems should be noted in the response (e.g., “too many records requested”, “search timed out”) so that predictable error-handling can be managed by the EDP. For further details see 8.2.6 in [5].

4.3.4.6 Service Information for Integration

To be able to integrate an OpenSearch(GEO/EOP) the following information need to be provided by the data supplier:

Service Information	Definition	Obligation (M=Mandatory, O=Optional, C=Conditional)	Datatype
URL Template	The OpenSearch URL template format can be used to represent a parameterized form of the URL by which a search engine is queried.	M	OSDD URL template format

	The EDP harvester client will process the URL template and attempt to replace the template parameters (as defined in Table 8). All template parameters must be instantiated, except for either startPage or startIndex, which are instatiated at runtime to iterate through the results.		

Table 9 Table of Service Information needed for Integration

Example URL template:

http://fedeo.esa.int/opensearch/request/?httpAccept=application/atom%2Bxml&parentIdentifier=EOP%3AESA%3AFEDEO&type=dataset%20series&recordSchema=iso&startRecord={startIndex?}

5 Operational requirements

5.1 Harvesting Frequency

Due to the high volume of metadata that will be harvested from a growing list of data suppliers and the required runtime for the harvesting processes, each data supplier site will probably not be harvested on a daily basis. Hereto the harvesting processes have to be clustered and scheduled on a fixed time schedule (e.g. during the night) in order to avoid any load impacts on the harvested sites during regular business hours usage by their users.

Log files which are written by the harvesting processes can be used to identify the optimal time to run the harvesters. In addition log-based evaluation methods will be used to monitor the performance of each harvester. If the performance of a harvester decreases due to resource problems, a ticket will be posted to the data source via the helpdesk.

5.2 Quality of the harvested datasets

5.2.1 Incremental/differential/selective harvesting

It must be clarified whether the harvested site supports incremental/differential and/or selective harvesting.

5.2.1.1 Metadata Timestamps

Hereto the datasets must include special data fields (e.g. a timestamp in order for the harvesting process to identify the date & time at which the dataset had been modified for the last time) based on which the harvester processes select only subsets of the metadatasets.

5.2.1.2 Data Source Site API / Endpoints

The REST API of the Data Source site should accept queries with e.g. startPos/maxRecs parameters for resumption / partitioning of the datasets to be harvested.

5.2.2 Avoiding duplicates

Duplicate datasets should be avoided by both the source site as well as by the the European Data Portal during harvesting.

5.2.3 Error reporting on harvested metadata

5.2.3.1 Jira Tickets

The harvester processes used by the European Data Portal report different types of errors encountered during harvesting of the datasets by issueing Jira tickets to the HelpDesk.

5.2.3.2 MQA-Metadata Quality Assurance

The MQA module provides a graphical report on the quality of the harvested datasets' metadata by providing access to a dachboard that summarizes the main quality indicators e.g. availability and accessibility of distributions, compliance of datasets to metadata formats, source of violations etc.

The MQA dashboard can be opened direcly from the Portal HomePage.

5.2.4 User feedback on datasets

Users will be able to provide feedback on a dataset directly from the dataset detail page.

The system will allow to gather and extract all feedback received for all datasets and group those by data supplier, so that the feedback can be sent to the data supplier.

6 Checklist

The goal of this checklist is to gather and summarize all main requirements for successfully harvesting a data supplier site and assure a certain quality level of the harvested datasets.

	Requirement	Value	Comment
1	Make sure that your portal provides metadata!		Only metadata can be harvested, not the data itself!
2	Which metadata standard is used?	CKAN/INSPIRE/DCAT-AP	If other, please provide a detailed description of the scheme used
3	Which representation of the metadata is used?	XML/JSON	
4	Which API is used to retrieve the data?	CKAN/OAI-PMH/dump file	
5	Is authentication required to access API?	yes/no	
6	Include complete vocabulary for categorization, or other fields that use a defined vocabulary (for example update frequency)		With translation, if applicable
7	Use standard date/time formats	ISO8601	
8	Is differential or incremental harvesting supported?	yes/no	Incremental or differential harvesting allows for harvesting of datasets that were changed after a certain date only
9	How often can/should the site be harvested?	daily/weekly/monthly/etc.	
10			

7 Questions

	Question	Answer
1		
2		
3		
4		
5		
6		
7		
8		
9		
10		

Annex A - General Requirements for File-based Resources of (Open) Datasets

7.1 Introduction

This section describes the technical requirements / recommendations for data formats of data resources that are part of / referenced by the harvested (open) datasets. The objective is to have a uniform base for data that will be made accessible and viewable by the European Data Portal.

The most commonly supported file-based data formats are:

- CSV (comma separated values)
- XLS(X) (Excel files)
- PDF (portable document format)
- RTF (rich text format)
- …

7.2 Standard requirements

- R1: Detailed Documentation

The structure of the data files that are part of datasets should be documented in detail, ideally on a dedicated page in the online data catalog.

- R2: Universal filenames

Filenames should only consist of small caps characters a-z, digits 0-9 and the underscore "_" character, this to ensure the processing of the data files on both server and user side.

- R3: Timestamp as part of the filename

Filenames should contain the date and/or time of their last update as suffix in the ISO-Format as follows:

[name]_YYYY-MM-DD

or [name]_YYYY-MM-DD-HH-MM

By providing the date/time of last update as part of the filename, the user can easily identify when the file has been modified without having to open the file.

Annex B: Recommendations for Dataset Resources in CSV format

CSV files are text files that should fulfill several structural criteria in order to allow for a simple and automatic processing of the data that they contain.

Standard requirements for CSV files are defined in RFC 4180 (http://tools.ietf.org/html/rfc4180).

7.3 CSV-1: Standard Character Set UTF-8

The CSV file should use the UTF-8 character set encoding.

In case that the CSV file is using another character set, this must be documented in the documentation.

7.4 CSV-2: CSV-File should only contain a single table of data

A CSV file should only consist of a table of data values that belong of the file.

Additional information on the data i.e. metadata like descriptions, comments, date of last update, etc. should not be included in the CSV file rather than described in the documentation and/or included in the file name.

In case that the additional information is requested to be part of the distribution and changes with each distribution, then a different format e.g. XML, JSON should be used instead of CSV.

Each CSV file should only contain a single table of data values. In case that multiple tables are required, each table should be included in a separate CSV file.

7.5 CSV-3: First row should contain the column headers

The first row of the CSV file should contain the column names.

A CSV file without the column names cannot be easily documented. In addition this obliges the user to “guess” and define own column names which could lead to confusion.

7.6 CSV-4: Column headers should use the universal text format

In order to be used efficiently used, the column names should follow the following structure:

- Only small caps letters a-z and digits 0-9 should be used
- No spaces should be included in the column names
- Separate words should be connected with the underscore (_) character
- No special characters should be used (e.g. äüöàéèê etc.)
- If possible the column names should be in English

7.7 CSV-5: Using the ";" as field separator character

The semi-colon character ";" should be used as field separator.

7.8 CSV-6: Use of double quote characters

Field values in text fields can be optionally enclosed in double quote characters (" "). This is mainly useful in case that spaces/special characters are part of the field value.

In case that the field value includes a double quote character (") then this character has to be duplicated i.e. applied twice.

e.g. `"This is a text that includes two ""double quote characters""."`

In case that the field value needs to include the field separator character (;) or a carriage return character, then the field value has to be enclosed in double quotes.

e.g. `"This is a text that includes the semi-colon ; character"`

7.9 CSV-7: Same number of colums in all rows

All rows in a CSV file should have the same number of columns i.e. all rows should have a field value for each column.

Empty columns should contain a double colon ":" character.

As a consequence all rows have the same number of field separators.

7.10 CSV-8: Only one data type per column

Data values in the same column should be of the same data type (e.g. text, integer, decimal, date, time, etc.).

In case of a column containing both integer and decimal values, the data type should be decimal.

7.11 CVS-9: Decimal Point

The dot-character (.) should be used as decimal point for decimal values only, as widely used in the English language area.

For columns of type integer, all integer values should be without decimal point and decimals.

This allows for a more compact data file and provides a visual identification of the column type as integer.

For columns of type decimal, all number values should include the decimal point and the same number of decimals (also for integer numbers).

7.12 CSV-10: Use of leading Zeros

Integer or decimal values should not use leading zeros.

In case that leading zeros are mandatory and need to be preserved (e.g. for codes) then the data type should be text and the data values should be enclosed in double quotes (“ “).

In case that text values need to include a double quote character, then this character needs to be doubled (e.g. “This text contains a ””double”” quote character”).

7.13 CSV-11: No “thousands” formatting characters

Number values (integers, decimals) can only consist of digits 0-9 and the optional “-“ character for negative numbers and the decimal point (.) for decimals.

They cannot contain formatting characters/separators e.g. “,” or blank for the “thousands” position or currency symbols e.g. €, £, $.

7.14 CSV-12: Use of Units/Measures

The units or measures for numeric values e.g. currency, km/h, etc. cannot be mixed with the numeric values in the same column.

In case that the unit/measure is the same for all values in a column, the unit can be added to the column name e.g. “`amount_eur`”.

In case that the unit/measure can have different values for a column, then the unit/measure must be stored in a separate column (ideally following the numeric value column).

Examples of units:

- ”description”;”amount”;”currency”
- ”Article-1”;123.45;”EUR”
- ”Article-2”;67.89;”USD”

7.15 CSV-13: Data values of type Date

For data values of type date the ISO-format should be used as follows:

YYYY-MM-DD

For data values of type date/time the **ISO-format** should be used as follows:

YYYY-MM-DD-HH-MM

or **YYYY-MM-DD-HH-MM-SS**

The weekday should not be included in the date value, nor in a separate column since it is redundant.

The hour character “h” or “H” should not be included in the date value.

For including a period or duration in the data file, two columns should be used: one for the start-date(time) and one for the end-date(time).

7.16 CSV-14: Assigning a unique ID

Each data row could be identified by a unique ID in the first column of the data file.

These IDs should:

- Not be changed in future updates of the data file:
  - Old data rows incl. their ID should be deleted from the data file,
  - Existing data rows should keep their ID,
  - New data rows should get a new unique ID.
- Not be re-used / re-cycled for new data rows.

7.17 CSV-15: Header row

As an alternative for the header row and similar to the Json-LD file format (http://json-ld.org), a URI pointing to the description of the format and content of the CSV file, could be used.

e.g. “#@context: http://aaa/bbb/fileformat.ld”.

Annex C: Recommendations for Dataset Resources in XLS(X) format

XLS(X) is a file extension for a spreadsheet file format created by Microsoft for use with Microsoft Excelthat and that should fulfill several structural criteria in order to allow for a simple and automatic processing of the data that they contain.

7.18 XLS(X)-1: Standard Character Set UTF-8

The XLS(X) file should use the UTF-8 character set encoding.

In case that the XLS(X) file is using another character set, this must be documented in the documentation.

7.19 XLS(X)-2: XLS(X)-File should only contain a table of data

A XLS(X) file should only consist of one or more spreadsheets with data values that belong to the file.

Additional information on the data i.e. metadata like descriptions, comments, date of last update, etc. should not be included in the XLS(X) file rather than described in the documentation and/or included in the file name.

In case that the additional information is requested to be part of the distribution and changes with each distribution, then a different format e.g. XML, JSON should be used instead of XLS(X).

Each XLS(X) spreadsheet should only contain a single table of data values. In case that multiple tables are required, each table should be included in a separate spreadsheet or in a different XLS(X) file.

7.20 XLS(X)-3: First row should contain the column headers

The first row of every table of data values in the XLS(X) spreadsheets should contain the column names.

A table of data without the column names cannot be easily documented. In addition this obliges the user to “guess” and define own column names which could lead to confusion.

7.21 XLS(X)-4: Column headers and spreadsheet names should use the universal text format

In order to be used efficiently used, the column and spreadsheet names should follow the following structure:

- Only small caps letters a-z and digits 0-9 should be used
- No spaces should be included in the column names
- Separate words should be connected with the underscore (_) character
- No special characters should be used (e.g. äüöàéèê etc.)

- If possible the column names should be in English

7.22 XLS(X)-5: Empty spreadsheets

Empty spreadsheets within a XLS(X) file must be avoided because they could lead to confusion.

7.23 XLS(X)-6: Use of double quote characters

Field values in text fields can be optionally enclosed in double quote characters (“ “). This is mainly useful in case that spaces/special characters are part of the field value.

In case that the field value includes a double quote character (“) then this character has to be duplicated i.e. applied twice.

e.g. `”This is a text that includes two ””double quote characters””.”`

In case that the field value needs to include the field separator character (;) or a carriage return character, then the field value has to be enclosed in double quotes.

e.g. `”This is a text that includes the semi-colon ; character”`

7.24 XLS(X)-7: Same number of colums in all rows

All rows in a XLS(X) table of data should have the same number of columns i.e. all rows should have a field value for each column.

Empty columns should contain a double colon “:” character.

As a consequence all rows have the same number of field separators.

7.25 XLS(X)-8: Only one data type per column

Data values in the same column should be of the same data type (e.g. text, integer, decimal, date, time, etc.).

In case of a column containing both integer and decimal values, the data type should be decimal.

7.26 XLS(X)-9: Decimal Point

The dot-character (.) should be used as decimal point for decimal values only, as widely used in the English language area.

For columns of type integer, all integer values should be without decimal point and decimals.

This allows for a more compact data file and provides a visual identification of the column type as integer.

For columns of type decimal, all number values should include the decimal point and the same number of decimals (also for integer numbers).

7.27 XLS(X)-10: Use of leading Zeros

Integer or decimal values should not use leading zeros.

In case that leading zeros are mandatory and need to be preserved (e.g. for codes) then the data type should be text and the data values should be enclosed in double quotes (“ “).

In case that text values need to include a double quote character, then this character needs to be doubled (e.g. “This text contains a ””double”” quote character”).

7.28 XLS(X)-11: No “thousands” formatting characters

Number values (integers, decimals) can only consist of digits 0-9 and the optional “-“ character for negative numbers and the decimal point (.) for decimals.

They cannot contain formatting characters/separators e.g. “,” or blank for the “thousands” position or currency symbols e.g. €, £, $.

7.29 XLS(X)-12: Use of Units/Measures

The units or measures for numeric values e.g. currency, km/h, etc. cannot be mixed with the numeric values in the same column.

In case that the unit/measure is the same for all values in a column, the unit can be added to the column name e.g. “`amount_eur`”.

In case that the unit/measure can have different values for a column, then the unit/measure must be stored in a separate column (ideally following the numeric value column).

7.30 XLS(X)-13: Data values of type Date

For data values of type date the ISO-format should be used as follows:

YYYY-MM-DD

For data values of type date/time the **ISO-format** should be used as follows:

YYYY-MM-DD-HH-MM

or **YYYY-MM-DD-HH-MM-SS**

The weekday should not be included in the date value, nor in a separate column since it is redundant.

The hour character “h” or “H” should not be included in the date value.

For including a period or duration in the data file, two columns should be used: one for the start-date(time) and one for the end-date(time).

7.31 XLS(X)-14: Assigning a unique ID

Each data row could be identified by a unique ID in the first column of the data file.

These IDs should:

- Not be changed in future updates of the data file:
  - Old data rows incl. their ID should be deleted from the data file,
  - Existing data rows should keep their ID,
  - New data rows should get a new unique ID.
- Not be re-used / re-cycled for new data rows.